



SERVIÇO PÚBLICO FEDERAL
UNIVERSIDADE FEDERAL DE SERGIPE
PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA

PROGRAMA INSTITUCIONAL DE BOLSAS DE INICIAÇÃO CIENTÍFICA – PIBIC

**Desenvolvimento de um controlador cinemático com
ponderação por parâmetros dinâmicos para robôs
manipuladores.**

Área do conhecimento: Engenharias
Subárea do conhecimento: Engenharia Eletrônica

Relatório Final
Período da bolsa: de Agosto de 2018 a Julho de 2019

Este projeto é desenvolvido com bolsa de iniciação científica
PIBIC/CNPq

Orientador: Elyson Ádan Nunes Carvalho
Autor: Raphael Cardoso de Oliveira Jesus

Sumário

1	Introdução	1
1.1	Objetivos	2
1.1.1	Objetivos Gerais	2
1.1.2	Objetivos Específicos	2
1.2	Estrutura do Relatório	2
2	Fundamentação Teórica	3
2.1	Manipuladores	3
2.2	Cinemática	4
2.2.1	Cinemática Direta	4
2.2.2	Cinemática Inversa	6
2.3	Dinâmica	7
2.3.1	O Lagrangiano	8
2.3.2	Modelo Dinâmico de um Manipulador	8
2.3.3	Jacobiano Dinamicamente Consistente	11
3	Trabalhos Relacionados	13
4	Metodologia para Implementação da Matriz de Inércia	15
4.1	Velocidade Linear	15
4.2	Velocidade Angular	16
4.3	A Matriz de Inércia	16
4.4	Cinemática Inversa com Ponderações Dinâmicas	17
5	Resultados e Discussões	20
5.1	Caso Planar	20
5.2	Caso Espacial	23
6	Considerações Finais	25
6.1	Atividades Complementares	26
	Referências Bibliográficas	27

Capítulo 1

Introdução

A área de manipuladores encontra grande aplicabilidade em vários campos do conhecimento, e, por conta disso, surge a necessidade de melhoramento contínuo por meio de pesquisa científica, mesmo sendo uma das mais antigas da robótica. Por ser tão bem estabelecida, essa área possui campos suficientemente aprofundados de forma que são pesquisados de forma quase independente.

Dentre os campos específicos da robótica de manipuladores, existe a Cinemática, que consiste em estudar as relações de posições e velocidades dos componentes do robô, sendo limitada, pois não consideram-se os torques envolvidos nos movimentos. Por isso, outra área importante no estudo de manipuladores é a Dinâmica. Nesse caso, são especificados torques ou acelerações desejadas para alguma parte do seu corpo, que serão executados por meio de torques gerados por motores. Esse processo envolve a solução de sistemas de equações diferenciais para encontrar a aceleração que deve ser executada por uma junta, por exemplo.

Dentro desse tema, Khatib [1] desenvolveu uma matriz chamada "Jacobiano Dinamicamente Consistente". A construção dessa se dá por uma composição entre pré e pós-multiplicação de matrizes determinadas por parâmetros inerciais sobre a Jacobiana, bem conhecida e explorada também na cinemática.

No controle cinemático, essa matriz é usada como uma forma de realizar a Cinemática Inversa, que consiste em traduzir velocidades de movimentação de qualquer ponto do robô para velocidades executadas nos motores. Métodos clássicos de Cinemática Inversa são baseados em inversões do Jacobiano [2].

A inversa da matriz Jacobiana do robô nem sempre é formalmente definida devido a regiões chamadas de singularidades. Por isso, existem várias alternativas que a modificam internamente, para garantir a existência da inversa. A forma mais bem-sucedida de realizar tal modificação é conhecida como *DLS (Damped Least Squares)* [3].

Como alternativa para as inversões Jacobianas, foi desenvolvido um método num trabalho anterior [4] que, baseado na pré-multiplicação da Jacobiana Transposta por uma matriz específica, conseguiu resultados promissores para manipuladores planares, mas sem

generalização direta para robôs espaciais. Por conta da semelhança de formato do Jacobiano Dinamicamente Consistente com o que foi desenvolvido no trabalho anterior, foi proposta a investigação dos efeitos dessa metodologia na Cinemática Inversa.

Assim, este trabalho visa implementar um algoritmo de Cinemática Inversa levando em consideração o estudo da Dinâmica de robôs manipuladores. Para realização dessas tarefas, são propostos os objetivos descritos na seção seguinte.

1.1 Objetivos

1.1.1 Objetivos Gerais

Este projeto tem como objetivo geral realizar controle cinemático de robôs manipuladores considerando aspectos relacionados à dinâmica do robô.

1.1.2 Objetivos Específicos

São objetivos específicos desse trabalho:

- Realizar um estudo de base sobre dinâmica de corpos rígidos e robôs manipuladores.
- Realizar uma revisão bibliográfica sobre técnicas de obtenção das matrizes de inércia;
- Implementar o controlador cinemático ponderado com as matrizes de inércia;
- Comparar os comportamentos de trajetórias e erro ao longo do tempo entre o método implementado, o método desenvolvido em trabalho anterior, e métodos clássicos da literatura;
- Divulgar resultados obtidos por meio de relatórios e artigos científicos.

1.2 Estrutura do Relatório

Este texto está disposto de acordo com a seguinte estrutura: no capítulo 2, fundamentos teóricos para o entendimento do trabalho são apresentados. No capítulo seguinte, é feito um resumo dos problemas abordados em artigos relacionados ao tema. O capítulo 4 descreve a implementação da metodologia escolhida para cálculo da matriz de inércia, além das discussões teóricas que fundamentaram o projeto do controlador. Os resultados dos testes realizados, bem como discussões a respeito desses, são mostrados no capítulo 5. No capítulo 6 são apresentadas as considerações finais e, ao final do documento, são encontradas as referências bibliográficas.

Capítulo 2

Fundamentação Teórica

Para o entendimento e desenvolvimento do trabalho, são necessários conhecimentos sobre as bases teóricas da robótica de manipuladores. Neste capítulo, serão abordados conceitos gerais, e então teorias mais específicas a respeito da cinemática e dinâmica.

2.1 Manipuladores

Um robô manipulador é composto por juntas, elos, e um ou mais efetadores, como caso de manipuladores paralelos. Os robôs abordados neste trabalho são manipuladores seriais sem ramificações, como o da Figura 2.1.

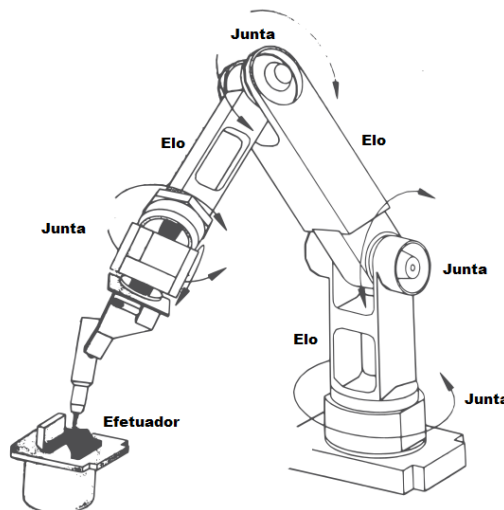


Figura 2.1: Representação de um manipulador serial.

A depender da tarefa, o robô pode ser adaptado de formas diferentes, tanto em relação ao efetador usado quanto em relação à construção do robô. No aspecto construtivo, podem ser usadas juntas de tipos distintos, trazendo diferentes opções de mobilidade para o manipulador.

As duas variações principais de juntas são: prismática e de revolução. Juntas prismáticas realizam movimento de translação de um elo em relação ao outro e em geral possuem equações mais simples. Já as juntas de revolução executam variações angulares entre elos. Como os modelos envolvendo juntas rotacionais são mais complexos e o robô disponível no laboratório é composto apenas por esse tipo de junta, este trabalho terá foco em juntas de revolução.

Além do tipo de juntas, a quantidade delas também é importante no modelo matemático do robô. Ao considerar juntas que só permitem movimento em uma direção (juntas simples), a quantidade de Graus de Liberdade (GdL) de um manipulador é igual à quantidade de juntas presentes em sua construção [5].

O processo de utilização de um manipulador envolve realização de movimentos de juntas com objetivo de executar uma tarefa com o efetuador. Para fazer isso com precisão, é necessário que exista algum tipo de sensoramento que determine, direta ou indiretamente a posição do robô no espaço.

Medições diretas são difíceis de serem realizadas e possuem aplicações limitadas. Por isso, a forma mais comum de sensoramento do robô é por meio de sensores localizados em cada junta. Para determinar então a posição de todo o robô baseado em medições de ângulo de juntas, é necessário conhecer as relações cinemáticas do robô.

2.2 Cinemática

Por definição, cinemática é a área que lida com o movimento de objetos (posições, velocidades) sem considerar as forças envolvidas no processo. Na robótica de manipuladores, essa área é dividida majoritariamente em Cinemática Direta e Inversa [6].

2.2.1 Cinemática Direta

Na Cinemática Direta, o objetivo é determinar a posição (\mathbf{p}) de qualquer parte do robô baseado nas medidas de variáveis de juntas (\mathbf{q}). Para isso, é comum definir um sistema de coordenadas para cada elo do manipulador. Assim, sabendo a relação entre os elos, é possível propagar a informação de qualquer ponto até a base do robô.

A relação entre dois sistemas de coordenadas é encontrada a partir da composição de dois movimentos: rotação e translação. Assim, supondo que existam dois sistemas de coordenadas, A e B, caso um ponto \mathbf{p} seja conhecido no sistema de coordenadas B, será denotado por \mathbf{p}^B . Para obter as coordenadas dele em relação ao sistema A, basta fazer

$$\mathbf{p}^A = T_B^A \mathbf{p}^B, \quad (2.1)$$

em que T_B^A é uma matriz de transformação que representa a conversão do sistema B para o A.

No caso de manipuladores, o ponto de interesse pode ter que passar por vários sistemas de coordenadas até chegar na base do robô. Matematicamente, esse encadeamento de transformações pode ser representado por uma multiplicação de matrizes da forma

$$T_n^0 = T_1^0 T_2^1 \dots T_n^{n-1}. \quad (2.2)$$

Essas matrizes T são particulares para cada sistema de coordenada, devendo ser encontradas individualmente. Por conta disso, projetistas diferentes podem obter sequências distintas de matrizes para o mesmo robô. Para mitigar esse problema e tornar o processo de obtenção das transformações mais sistemático foi proposta a convenção de Denavit-Hartenberg (D-H).

Essa convenção foi pensada para definir sistemas de coordenadas entre elos vizinhos, considerando que o eixo de coordenadas i equivale ao i -ésimo elo da estrutura. Para encontrar cada uma das transformações T_i^{i-1} , a convenção D-H primeiro define duas regras para posicionamento dos eixos de coordenadas.

- DH1:** O eixo X_i deve ser perpendicular ao eixo Z_{i-1} ;
DH2: O eixo X_i deve interceptar o eixo Z_{i-1} .

Com **DH1** e **DH2** satisfeitas, a relação entre os eixos (X_i, Y_i, Z_i) e $(X_{i-1}, Y_{i-1}, Z_{i-1})$ é dada por

$$T_i^{i-1} = Rot_{z, \theta_i} Trans_{z, d_i} Trans_{x, a_i} Rot_{x, \alpha_i} \quad (2.3)$$

$$= \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

em que senos e cossenos dos ângulos são representados pelas letras s e c , respectivamente.

A matriz obtida depende de quatro parâmetros, denominados parâmetros Denavit-Hartenberg, sendo:

- a_i = distância de Z_{i-1} para Z_i , medida ao longo de X_{i-1} ;
- α_i = ângulo entre Z_{i-1} e Z_i , medido em torno de X_{i-1} ;
- d_i = distância de X_{i-1} para X_i , medida ao longo de Z_{i-1} ;
- θ_i = ângulo entre X_{i-1} e X_i , medido em torno de Z_{i-1} .

No caso de juntas rotacionais, o termo que varia com o movimento da junta é θ_i , sendo também denominado variável generalizada da junta q_i .

Com uma representação cinemática consistente obtida através da convenção D-H, é possível localizar qualquer ponto do robô conhecendo os parâmetros D-H fixos (que dependem da estrutura) e das variáveis de junta para cada tempo. No entanto, o processo de movimentação de um robô requer que, dado um ponto no espaço, se encontrem as variáveis de junta que o levem para a posição desejada, caracterizando o problema de Cinemática Inversa, que pode ser realizada com foco em qualquer ponto do robô, mas neste trabalho, o ponto de interesse é o efetuador.

2.2.2 Cinemática Inversa

A Cinemática Direta pode ser vista como uma função cujo domínio é o Espaço de Configuração (variáveis de juntas), e sua imagem é o Espaço de Trabalho (posições espaciais). Assim, a função cinemática é descrita por

$$\mathbf{p} = f(\mathbf{q}), \quad (2.4)$$

em que variáveis escritas em negrito representam vetores.

Para realizar a Cinemática Inversa, então, seria necessário algum tipo de função inversa à f . Isto, no entanto, não é possível pois existem diferentes configurações (conjuntos de \mathbf{q}) que levam à mesma posição \mathbf{p} . Uma solução é transformar o problema de posições em um problema de velocidades, linearizando localmente a relação (2.4), obtendo

$$\dot{\mathbf{p}} = J(\mathbf{q})\dot{\mathbf{q}}. \quad (2.5)$$

Para determinar a matriz Jacobiana, há duas formas comuns: realizar as derivadas das equações cinemáticas em termos de cada variável de junta (Jacobiano Analítico) ou aproveitar de propriedades de matrizes de rotação e matrizes antissimétricas (Jacobiano Geométrico) [5].

Assim, caso a matriz Jacobiana $J(\mathbf{q})$ possua inversa, um vetor de velocidades para variáveis de junta pode ser encontrado fazendo $J^{-1}(\mathbf{q})\dot{\mathbf{p}} = \dot{\mathbf{q}}$. Ao realizar esse procedimento de forma iterativa, por meio de integração no tempo, é possível chegar ao vetor \mathbf{q} que leva o manipulador à posição desejada.

Essa abordagem é simples, porém muito limitada. Por exemplo, uma matriz só possui inversa se for quadrada, mas a dimensão de J é de $m \times n$, em que n é o número de graus de liberdade e m depende da tarefa (dimensão de \mathbf{p}). Para solucionar isso, a pseudoinversa de J pode ser usada, denotada por J^\dagger .

$$J^\dagger = (J^T J)^{-1} J^T. \quad (2.6)$$

Outro problema é que, como J depende da configuração atual do robô, é possível que, em alguma região, a inversa (ou pseudoinversa) não seja bem definida. Essas regiões são

denominadas de singularidades, pois a matriz Jacobiana se torna singular.

Uma matriz singular possui algum autovalor como zero. Logo, uma forma de mitigar o problema é somar uma pequena quantia λ aos autovalores da Jacobiana. Essa metodologia é conhecida como *Damped Least Squares (DLS)* e pode ser aplicada fazendo

$$J_{DLS} = (J^T J + \lambda I)^{-1} J^T. \quad (2.7)$$

Outra forma de evitar problemas de inversão é aproximar J^{-1} por J^T . Essa aproximação seria ideal se a Jacobiana fosse ortogonal, condição em que de fato a inversa é igual à transposta. Apesar desse tipo de solução não sofrer com regiões de singularidade, fazer essa aproximação cria soluções muito instáveis, apesar de rápidas.

As metodologias descritas previamente transformam o problema de Cinemática Inversa num problema de inversão matricial, mas também é possível realizar essa tarefa de outras formas: soluções analíticas, redes neurais, algoritmos genéticos e filtros de partículas, que estão fora do escopo deste trabalho.

Com as cinemáticas Direta e Inversa, é possível realizar a operação de um robô manipulador sem considerar as forças e energias envolvidas. Como no escopo desse trabalho se consideram aspectos dinâmicos para fazer controle cinemático, se torna necessário um estudo básico sobre Dinâmica.

2.3 Dinâmica

Equações dinâmicas explicitamente relacionam forças ao movimento, ao contrário dos modelos cinemáticos, que focam apenas no movimento. Sendo um corpo físico real, um robô é composto por virtualmente infinitas partículas, então para modelar sua dinâmica surgem infinitas variáveis de posição a serem determinadas.

Felizmente, existem formas de simplificar o modelo para um sistema de corpos rígidos. Um corpo rígido é aquele cujas partículas possuem posições relativas fixas, mantidas por forças internas, chamadas forças de vínculo, que não realizam trabalho. Considerando as forças de vínculo como restrições num sistema tridimensional, restam seis variáveis: três cartesianas para descrever a localização do objeto e três ângulos para especificar sua rotação.

É importante mencionar que nenhum sólido é perfeitamente rígido, mas na maioria dos problemas não há diferença significativa ao considerá-lo como tal. Apesar disso, existem estudos específicos para incluir imperfeições no modelo de corpos rígidos, como braços flexíveis [7].

Este trabalho utiliza o modelo de corpos rígidos para manipuladores. Por isso, cada elo do robô pode ser visto como um corpo rígido que deve ser modelado por uma massa (M) posicionada no ponto de centro de massa da estrutura, e uma inércia (I) em torno

deste.

Para sistemas simples, como manipuladores de poucas juntas, é possível utilizar a aproximação de corpos rígidos em conjunto com a Segunda Lei de Newton ($F = m\ddot{q}$) para descrever a dinâmica completamente. No entanto, a falta de escalabilidade dessa abordagem cria a necessidade de uma metodologia mais genérica para modelar manipuladores. Essa metodologia muito bem estabelecida no estudo de mecânica clássica é conhecida como equações de Euler-Lagrange, baseadas no Lagrangiano.

2.3.1 O Lagrangiano

A ideia de Lagrange ao desenvolver esta metodologia foi de estabelecer equações de movimento de maneira genérica, baseada na observação da energia do sistema. Além disso, nem sempre é conveniente representar o sistema em coordenadas cartesianas, por isso, essa metodologia é construída sobre coordenadas generalizadas.

Coordenadas generalizadas, denotadas por q , são aquelas que representam completa e independentemente um sistema de partículas. No caso de robôs manipuladores, essas coordenadas são as variáveis de juntas.

As relações de forças, energias e momentos são análogas entre coordenadas cartesianas e generalizadas. A diferença é que modelos usando coordenadas generalizadas podem ser usados para representar o mesmo sistema de forma mais simples. Um exemplo disso é quando se utilizam coordenadas polares para representar funções no eixo cartesiano.

Considerando um sistema de n coordenadas generalizadas, a dinâmica de um sistema pode ser descrita por n equações do tipo

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \tau_i. \quad (2.8)$$

Na equação (2.8), τ_i é um componente de força externa e L é o Lagrangiano, definido como $L = K - P$: a diferença entre energias cinética e potencial.

Essa equação é similar à Segunda Lei de Newton, mas segue uma abordagem diferente que, no caso de manipuladores, facilita a análise dinâmica no sentido de que o problema se torna descrever as energias do sistema, permitindo compor mais facilmente o modelo de um manipulador.

2.3.2 Modelo Dinâmico de um Manipulador

Considerando a abordagem com o Lagrangiano, é necessário calcular inicialmente as energias cinética e potencial do sistema e aplicá-las na equação (2.8) para encontrar o modelo do robô.

2.3.2.1 Energia Cinética

Como o modelo considerado é de corpos rígidos, para cada elo, a energia cinética K é a soma de componentes lineares e angulares, dados por

$$K = \frac{1}{2}M\mathbf{v}^T\mathbf{v} + \frac{1}{2}\mathbf{w}^T I' \mathbf{w}, \quad (2.9)$$

em que \mathbf{v} e \mathbf{w} são respectivamente as velocidades linear e angular do corpo.

Nessa equação, o termo I' é uma matriz 3×3 chamada de tensor de inércia no sistema global de coordenadas. Como no modelo de manipuladores há um sistema de coordenadas para cada elo, cada tensor de inércia possui sua representação num sistema local. Essa representação é dada por I , que é a definição padrão de inércia de um corpo rígido, calculada como

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}.$$

Cada termo depende da densidade de massa ρ e formato do corpo. Os elementos da diagonal principal são os momentos principais de inércia e são calculados de forma análoga:

$$I_{xx} = \iiint (y^2 + z^2) \rho(x, y, z) dx dy dz. \quad (2.10)$$

Os termos restantes são os momentos de inércia cruzados, calculados de forma também análoga:

$$I_{xy} = I_{yx} = - \iiint xy \rho(x, y, z) dx dy dz. \quad (2.11)$$

Ao calcular a representação local do tensor de inércia, esta pode ser transportada para o sistema global fazendo

$$I' = R^T I R. \quad (2.12)$$

Assim, a energia cinética total se torna

$$K = \sum_{i=1}^n \frac{1}{2} M_i \mathbf{v}_i^T \mathbf{v}_i + \frac{1}{2} \mathbf{w}_i^T R_i^T I_i R_i \mathbf{w}_i. \quad (2.13)$$

Separando a equação (2.5) em componentes de velocidade linear e angular, surgem os Jacobianos J_v e J_w , respectivamente,

$$\mathbf{v}_i = J_{v_i}(\mathbf{q}) \dot{\mathbf{q}},$$

$$\mathbf{w}_i = J_{w_i}(\mathbf{q}) \dot{\mathbf{q}},$$

então a equação de energia cinética total é expandida para

$$K = \frac{1}{2} \dot{\mathbf{q}}^T \sum_{i=1}^n (M_i J_{v_i}^T J_{v_i} + J_{w_i}^T R_i^T I_i R_i J_{w_i}) \dot{\mathbf{q}} = \frac{1}{2} \dot{\mathbf{q}}^T D \dot{\mathbf{q}}. \quad (2.14)$$

O termo da soma resulta numa matriz $n \times n$, sendo que n é o número de juntas. Essa matriz é denominada Matriz de Inércia no Espaço de Configuração e denotada por D , e possui como propriedades o fato de ser simétrica e positiva definida [5].

Essa formulação de energia cinética é feita considerando o Espaço de Configuração. Existe uma formulação alternativa no Espaço de Trabalho, dada por

$$K = \dot{\mathbf{p}}^T \Lambda \dot{\mathbf{p}}. \quad (2.15)$$

A matriz Λ é chamada de Matriz de Inércia no Espaço de Trabalho e se relaciona com D por $\Lambda = J^{T-1}(\mathbf{q}) D J^{-1}(\mathbf{q})$ [1].

Por meio de (2.14), é possível calcular a energia cinética de forma algorítmica, com base nos Jacobianos.

2.3.2.2 Energia Potencial

No caso de corpos rígidos, a única fonte de energia potencial é a gravidade. Portanto, para calcular a energia, basta saber a massa do objeto (M_i), a direção da gravidade (\mathbf{g}), e a posição do elo (\mathbf{r}_i).

Assim,

$$P = \sum_{i=1}^n \mathbf{g}^T \mathbf{r}_i M_i. \quad (2.16)$$

É importante notar que P só depende da posição do objeto no espaço, e não de sua velocidade. Com P e K definidos, a formulação do Lagrangiano pode ser utilizada.

2.3.2.3 Equação de Movimento

Após calcular $L = K - P$ e aplicar em (2.8), as n equações de Euler-Lagrange podem ser escritas no formato matricial [5] como

$$D(\mathbf{q}) \ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + g(\mathbf{q}) = \boldsymbol{\tau}. \quad (2.17)$$

Nessa representação, há a Matriz de Inércia no Espaço de Configuração (D), a matriz C que possui derivadas dos elementos de D , e o vetor de elementos provenientes da energia potencial $g(\mathbf{q})$. Ao resolver essa equação diferencial, é possível relacionar torques a acelerações no Espaço de Configuração. No entanto, da mesma forma que na Cinemática

Inversa, é importante descrever relações entre os espaços de Configuração e Trabalho, por isso, existe a equação de movimento no Espaço de Trabalho, dada por

$$\Lambda(\mathbf{x})\ddot{\mathbf{x}} + \mu(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}} + p(\mathbf{x}) = F. \quad (2.18)$$

Para evitar confusão com o termo de energia potencial p , a variável de posição foi substituída por \mathbf{x} , Λ é a matriz definida em (2.15), e μ é o termo com as derivadas de Λ . Devido à similaridade estrutural entre os dois modelos, uma relação entre cada um dos termos é esperada. De fato, uma relação importante é entre as forças τ e F

$$F = J^T \tau. \quad (2.19)$$

A associação entre torques no espaço de trabalho (F) e no espaço de configuração (τ) é muito relevante pois fornece o mapeamento necessário para garantir que os limites de esforço nos atuadores não sejam excedidos. A relação (2.19), no entanto só é válida no caso de robôs não redundantes, ou seja, quando a dimensão do espaço de configuração é a mesma do espaço de trabalho. Buscando um equacionamento genérico para relacionar espaço de trabalho e de configuração, Khathib [1] desenvolveu o conceito de Jacobiano Dinamicamente Consistente.

2.3.3 Jacobiano Dinamicamente Consistente

A necessidade dessa matriz surge ao incrementar o número de graus de liberdade a ponto do robô se tornar redundante, em que mais padrões de movimento se tornam possíveis, de forma que o manipulador pode ajustar sua postura sem afetar o posicionamento do efetuador. Esse tipo de movimento é chamado de movimento no espaço nulo do Jacobiano e deve ser incorporado na equação de forças para torná-la genérica, resultando no segundo termo de (2.20).

$$\tau = J^T F + (I_d - J^T J^{T\#})\tau_0, \quad (2.20)$$

em que I_d é a matriz identidade, τ_0 é uma constante e $J^{T\#}$ é uma aproximação da inversa de J^T .

Substituindo essa expressão para τ na equação diferencial do Espaço de Configuração (2.17), e fazendo a substituição

$$\ddot{\mathbf{x}} = \dot{J}\dot{\mathbf{q}} + J\ddot{\mathbf{q}},$$

obtém-se uma relação entre aceleração do efetuador ($\ddot{\mathbf{x}}$) e velocidades de juntas, dada por

$$\ddot{\mathbf{x}} + JD^{-1}g(\mathbf{q})\dot{\mathbf{q}} - \dot{J}\dot{\mathbf{q}} + JD^{-1}g(\mathbf{q}) = (JD^{-1}J^T)F + JD^{-1}(I - J^T J^{T\#})\tau_0. \quad (2.21)$$

Cada termo de (2.21) é uma fonte de aceleração, e para garantir que o torque gerado pelo movimento no espaço nulo não gere aceleração de efetuador, o termo $JD^{-1}(I - J^T J^{T\#})\tau_0$ deve ser zero. Rearranjando os termos, se obtém

$$J^\# = D^{-1} J^T (JD^{-1} J^T)^{-1}. \quad (2.22)$$

Esse resultado foi chamado de Jacobiano Dinamicamente Consistente, e é utilizado para mapeamento de torques entre os espaços de configuração e de trabalho.

Capítulo 3

Trabalhos Relacionados

Em estudos iniciais de cinemática inversa de manipuladores, Whitney [8] propôs que ao invés de controlar o robô por $\dot{\mathbf{q}} = J^{-1}\dot{\mathbf{p}}$, fosse feito

$$\dot{\mathbf{q}} = W^{-1}J(JW^{-1}J^T)^{-1}\dot{\mathbf{p}}, \quad (3.1)$$

em que W é uma matriz de ponderação arbitrária.

Esse equacionamento minimiza o termo $\dot{\mathbf{q}}^T W \dot{\mathbf{q}}$, que pode ser visto como uma função de custo, podendo produzir um comportamento mais robusto a singularidades [9], ou até desempenhar outras tarefas. Deve-se notar a similaridade dessa formulação com (2.22). De fato, o Jacobiano Dinamicamente Consistente é um caso específico em que $W = D$ e o termo minimizado é $\dot{\mathbf{q}}^T D \dot{\mathbf{q}}$, a energia cinética.

Após desenvolvimento do conceito de Jacobiano Dinamicamente Consistente por Khattib [1], surgiram aplicações dessa matriz para o controle dinâmico de robôs redundantes dentro do espaço nulo com robustez a singularidades [10]. Ao utilizar $J^\#$, da equação (2.22), foi possível garantir que torques no espaço de juntas não causariam acelerações de efetuador, mantendo-o em repouso.

Essa nova matriz se tornou relativamente popular ao longo do tempo no controle por espaço nulo [11], sendo generalizado para manipuladores móveis [12] e até humanoides [13]. Uma aplicação recente do Jacobiano Dinamicamente consistente em cinemática consiste numa metodologia baseada em dinâmica para formulação do Jacobiano Estendido [14]. Essa formulação é resultado de um modelo analítico do robô, ou seja, precisa ser reformulada para uso em robôs diferentes.

Por outro lado, no trabalho desenvolvido anteriormente [4], a Cinemática Inversa foi realizada para manipuladores planares baseando-se na relação

$$\dot{\mathbf{q}} = kP_e J^T \dot{\mathbf{p}}, \quad (3.2)$$

em que k é uma constante definida como $k = \frac{1}{2\sum_{i=1}^n L_i^2}$ e P_e é uma matriz simétrica e positiva definida que depende do número de juntas. No caso de um robô de três juntas,

$$P_e = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}.$$

Essa metodologia apresentou robustez a singularidades, mas não foi generalizada de forma direta para manipuladores espaciais. Em [15], foi desenvolvida uma solução que se baseia em transformar o robô espacial em três manipuladores planares, realizando a Cinemática Inversa independentemente usando o método da Pré-Multiplicação, e então compor o movimento dos três planos para executá-lo no robô.

Este trabalho visa desenvolver uma metodologia genérica, sem necessidade de reformulação com troca de modelo, da cinemática inversa baseada na equação (3.2) e nos conceitos de dinâmica, inspirado pelo Jacobiano Dinamicamente Consistente.

Capítulo 4

Metodologia para Implementação da Matriz de Inércia

Neste capítulo, é discutida a implementação de uma metodologia genérica para encontrar a Matriz de Inércia no Espaço de Configuração com objetivo de utilizá-la para controle cinemático. Para garantir o caráter genérico do método, todas as etapas dependem apenas do modelo de Denavit-Hartenberg do robô e são calculadas numericamente.

De acordo com a equação

$$D = \sum_{i=1}^n (M_i J_{v_i}^T J_{v_i} + J_{w_i}^T R_i^T I_i R_i J_{w_i}), \quad (4.1)$$

o cálculo de D depende de dois termos: de velocidade linear e angular. Cada um deles envolve um tipo diferente de matriz Jacobiana que deve ser encontrada. Nas seções seguintes serão mostradas as abordagens utilizadas para calcular cada termo da Matriz de Inércia, as simplificações realizadas no processo, e como utilizá-la para Cinemática Inversa.

4.1 Velocidade Linear

Antes de calcular o Jacobiano de fato, é importante lembrar que devem ser calculadas n matrizes Jacobianas para encontrar a Matriz de Inércia, ao invés de apenas uma (Jacobiano do efetuador), que é o procedimento comum na maioria dos métodos. Além disso, a origem do sistema de coordenadas de cada elo, para o cálculo do Jacobiano, deve se localizar no seu centro de massa.

De maneira geral, a determinação da matriz Jacobiana depende apenas dos quatro parâmetros D-H de cada sistema de coordenadas. Com isso, a coluna j do i -ésimo Jacobiano

de Velocidade Linear é dada por

$$z_{j-1} \times (o_i - o_{j-1}), \quad (4.2)$$

em que o_i é a posição do centro de massa, único para cada Jacobiano, o_{j-1} é a posição da junta anterior ao centro de massa, e z_{j-1} é um versor que representa a direção de atuação da junta, calculado como $z_{j-1} = T_{j-1}^0[0, 0, 1]^T$.

Ao finalizar o processo para todos os centros de massa, seriam obtidos Jacobianos de dimensões diferentes, pois para cada etapa há uma quantidade diferente de juntas a serem consideradas.

Para tornar a representação consistente, sabe-se que a dimensão do Jacobiano relativo ao efetuador é $m \times n$. Portanto, as matrizes devem ser completadas com zeros nas colunas à direita, para que todas tenham mesmas dimensões. Ao final desse procedimento, devem ser obtidos n Jacobianos de Velocidade Linear.

4.2 Velocidade Angular

O Jacobiano de Velocidade Angular é mais simples que o Linear, pois a velocidade angular ao longo de um corpo rígido é constante, ou seja, não é necessário considerar a posição do centro de massa. Assim, a coluna j do Jacobiano de Velocidade Angular para a i -ésima junta é, simplesmente, z_{j-1} [5].

Da mesma forma que explicitado na seção anterior, as matrizes devem ser completadas para manter a consistência de dimensões. É importante lembrar que a definição do Jacobiano depende da posição atual do manipulador, portanto os Jacobianos devem ser recalculados a cada movimento.

4.3 A Matriz de Inércia

De acordo com a definição em (2.14), a Matriz de Inércia no Espaço de Configuração é dada por

$$D = \sum_{i=1}^n (M_i J_{v_i}^T J_{v_i} + J_{w_i}^T R_i^T I_i R_i J_{w_i}). \quad (4.3)$$

Para completar o cálculo, basta avaliar os termos M e I , considerando que as matrizes R_i são rotações que levam do sistema de coordenadas local de cada elo para o sistema global. Como o foco do trabalho é em cinemática, a modelagem dinâmica do robô pode ser simplificada. Portanto, considera-se uma distribuição simétrica de massas.

Quando há simetria num elo, a matriz I_i é diagonal e constante, podendo ser simpli-

ficada por

$$I_i = \begin{bmatrix} k_{I_i} & 0 & 0 \\ 0 & k_{I_i} & 0 \\ 0 & 0 & k_{I_i} \end{bmatrix} = k_{I_i} I_d. \quad (4.4)$$

em que o termo k_{I_i} é um número real que representa o valor de inércia do elo e, I_d é a matriz identidade. Neste caso, o termo de velocidade angular $J_{w_i}^T R_i^T I_i R_i J_{w_i}$ se torna

$$J_{w_i}^T R_i^T (k_{I_i} I_d) R_i J_{w_i}. \quad (4.5)$$

Como a matriz identidade é o elemento unitário do produto matricial e o termo k_{I_i} é um número real, o termo indicado em (4.5) pode ser reescrito como

$$k_{I_i} (J_{w_i}^T R_i^T R_i J_{w_i}). \quad (4.6)$$

Aproveitando-se do fato de que matrizes de rotação são ortogonais, $R_i^T R_i = I_d$, permitindo que o termo de velocidade angular seja simplificado para

$$k_{I_i} J_{w_i}^T J_{w_i}. \quad (4.7)$$

Assim, a Matriz de Inércia no Espaço de Configuração simplificada, considerando um robô de estrutura simétrica, é dada por

$$D = \sum_{i=1}^n (M_i J_{v_i}^T J_{v_i} + k_{I_i} J_{w_i}^T J_{w_i}). \quad (4.8)$$

Desta forma, é possível calcular D e Λ , formando o Jacobiano Dinamicamente Consistente, que pode ser investigado como forma de realização de cinemática inversa.

4.4 Cinemática Inversa com Ponderações Dinâmicas

Como o interesse é puramente cinemático, os parâmetros dinâmicos de fato (massas e inércias) podem ser consideradas constantes de controlador, usadas para modificar o comportamento do robô. Dessa forma, de acordo com a formulação de Whitney [8], realizar a cinemática inversa da utilizando

$$\dot{\mathbf{q}} = D^{-1} J^T (J D^{-1} J^T)^{-1} \dot{\mathbf{p}} = D^{-1} J^T \Lambda \dot{\mathbf{p}}, \quad (4.9)$$

o termo $\dot{\mathbf{q}}^T D \dot{\mathbf{q}}$ é minimizado, como sugere Hollerbach [9], resultando na mesma formulação do Jacobiano Dinamicamente Consistente. Apesar disso, a inclusão do termo $(J D^{-1} J^T)^{-1}$ torna a cinemática inversa novamente sensível a singularidades.

Neste trabalho, foi proposta a remoção do termo de pós multiplicação, realizando a cinemática inversa apenas com

$$\dot{\mathbf{q}} = D^{-1} J^T \dot{\mathbf{p}}. \quad (4.10)$$

Dessa forma, a única inversão matricial restante é a de D , que não gera problemas pois é uma matriz positiva definida [5]. Portanto, D^{-1} se torna uma escolha baseada no modelo dinâmico para uma matriz de pré-multiplicação do Jacobiano Transposto.

A seguir, é apresentado um exemplo de construção da matriz D para um robô planar a fim de ilustrar o significado da Matriz de Inércia num contexto simplificado.

De acordo com a equação (4.8), os termos relativos a velocidades angulares podem ser independentemente calculados como

$$D_w = \sum_{i=1}^n k_{I_i} J_{w_i}^T J_{w_i}. \quad (4.11)$$

Para um manipulador planar, o Jacobiano Angular de cada junta é uma matriz constante. No caso de três juntas ($n = 3$), tem-se que

$$J_{w_1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, J_{w_2} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}, J_{w_3} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}. \quad (4.12)$$

Fazendo todos $k_{I_i} = 1$, a parte angular de D , é

$$D_w = \begin{bmatrix} 3 & 2 & 1 \\ 2 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \quad (4.13)$$

Voltando à equação total da matriz de inércia (4.8), repetida por legibilidade,

$$D = \sum_{i=1}^n (M_i J_{v_i}^T J_{v_i} + k_{I_i} J_{w_i}^T J_{w_i}),$$

nota-se que D é a soma de D_w com um termo que depende das velocidades lineares e da massa. Se considerarmos $M_i = 0 \forall i$, D se torna D_w , e a cinemática inversa pode ser realizada de acordo com (4.10), bastando calcular a inversa de D ,

$$D^{-1} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}. \quad (4.14)$$

Com isso, chega-se a um importantíssimo resultado: o equacionamento realizado a partir da operação reversa de redes neurais que resultou numa matriz de Pré-Multiplicação

[4], que é exclusivo a robôs planares, equivale a utilizar a Matriz de Inércia no caso de massa zero.

Ao adicionar novamente a massa ao sistema, D^{-1} continua do mesmo formato, porém os valores se modificam conforme a configuração.

No próximo capítulo, são mostrados os testes realizados tanto no caso planar, para ilustrar o que acontece com as trajetórias ao modificar os parâmetros de inércia, tanto no caso espacial, para verificar se o algoritmo é uma forma válida de cinemática inversa com robustez a singularidades.

Capítulo 5

Resultados e Discussões

Neste capítulo são mostrados resultados por simulação da aplicação do controlador cinemático desenvolvido em diferentes situações e modelos de robôs, comparando trajetórias e rapidez de convergência com outros algoritmos. O objetivo principal dos testes foi comparar as diferenças entre o método desenvolvido neste trabalho com o que resultou do trabalho anterior: a matriz de Pré-Multiplicação, para então verificar como a ponderação pela Matriz de Inércia se propaga para o caso espacial.

Na seção seguinte, são mostrados testes realizados num robô planar, visualizando também a interferência qualitativa dos parâmetros. Na seção posterior, são discutidos resultados obtidos com um manipulador espacial similar ao existente no Laboratório de Pesquisa em Robótica.

5.1 Caso Planar

No caso planar, o robô utilizado é de quatro juntas, tal que o comprimento de todos os elos é unitário. Os experimentos foram realizados a partir de uma posição inicial completamente vertical $\mathbf{p} = [0, 4]^T$, que corresponde uma posição de singularidade. A posição de destino considerada foi $\mathbf{pDes} = [2, 0]^T$.

Neste teste, foram implementadas quatro formas de obter $\dot{\mathbf{q}}$, e foi utilizado o método de integração de Euler para obter o próximo valor de q , conforme indicado em (5.1).

$$\mathbf{q}(i+1) = \mathbf{q}(i) + k\dot{\mathbf{q}}(i). \quad (5.1)$$

A constante k , em conjunto com os parâmetros específicos dos métodos implementados, está representada na Tabela 5.1. A respeito do método implementado neste trabalho, foram utilizadas duas variações de parâmetros de inércia dos elos, denominados Inércia 1 e 2.

Na Figura 5.1, são mostrados os resultados de todos os métodos para esse conjunto de posições inicial e final.

Tabela 5.1: Parâmetros utilizados em cada método.

	Pré-Multiplicação	DLS	Inércia 1	Inércia 2
k	$1/(2 \sum_{i=1}^n L_i^2)$	$1/(2 \sum_{i=1}^n L_i^2)$	$1/(2 \sum_{i=1}^n L_i^2)$	$1/(2 \sum_{i=1}^n L_i^2)$
M_i	-	-	0,1	0,1
k_I	-	-	$[1, 1, 1, 1]^T$	$[1, 1, 2, 3]^T$
λ	-	0,1	-	-

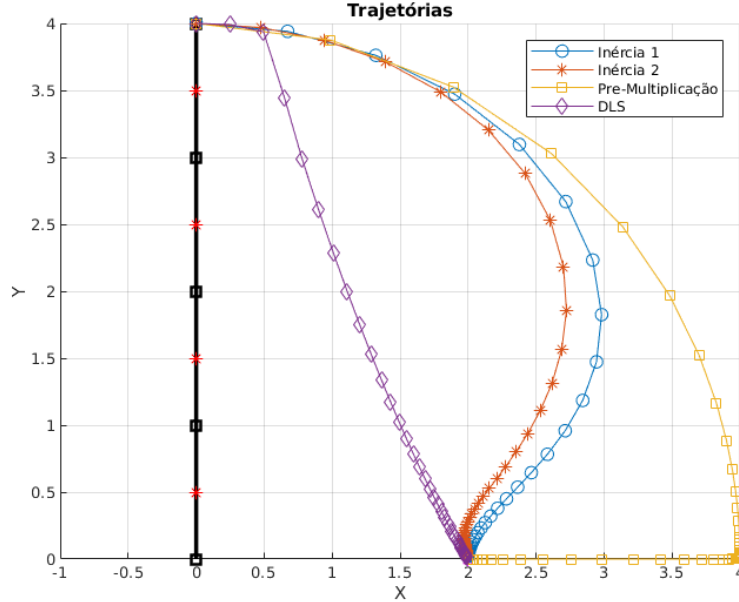


Figura 5.1: Trajetórias executadas pelo robô planar. Em preto, a configuração inicial do robô, e em asteriscos vermelhos sobre seu corpo, os centros de massa de cada elo.

No gráfico, é possível observar a dificuldade que o algoritmo da Pré-Multiplicação enfrenta tipo de situação, o que não acontece no caso da utilização da Matriz de Inércia. Além disso, as trajetórias do efetuador mostradas nas curvas Inércia 1 e 2 são muito similares, mas apresentaram padrões de movimento muito diferentes, que podem ser melhor observados nos gráficos da Figura 5.2.

Observando os valores do ângulo de cada junta é possível notar que o corpo do robô se moveu de duas formas distintas. Por exemplo, a curva em amarelo (q3) termina negativa no primeiro caso, enquanto no segundo, seu valor final é positivo.

A respeito das velocidades das juntas, é possível notar que no segundo caso o controlador priorizou movimento da terceira junta (pico em \dot{q}_3) e quase não moveu a quarta junta.

Em termos de velocidade de convergência, os algoritmos apresentaram resultados não muito distintos, exceto pela Pré-Multiplicação. No gráfico da Figura 5.3 é ilustrada a norma do erro do efetuador ao longo do tempo.

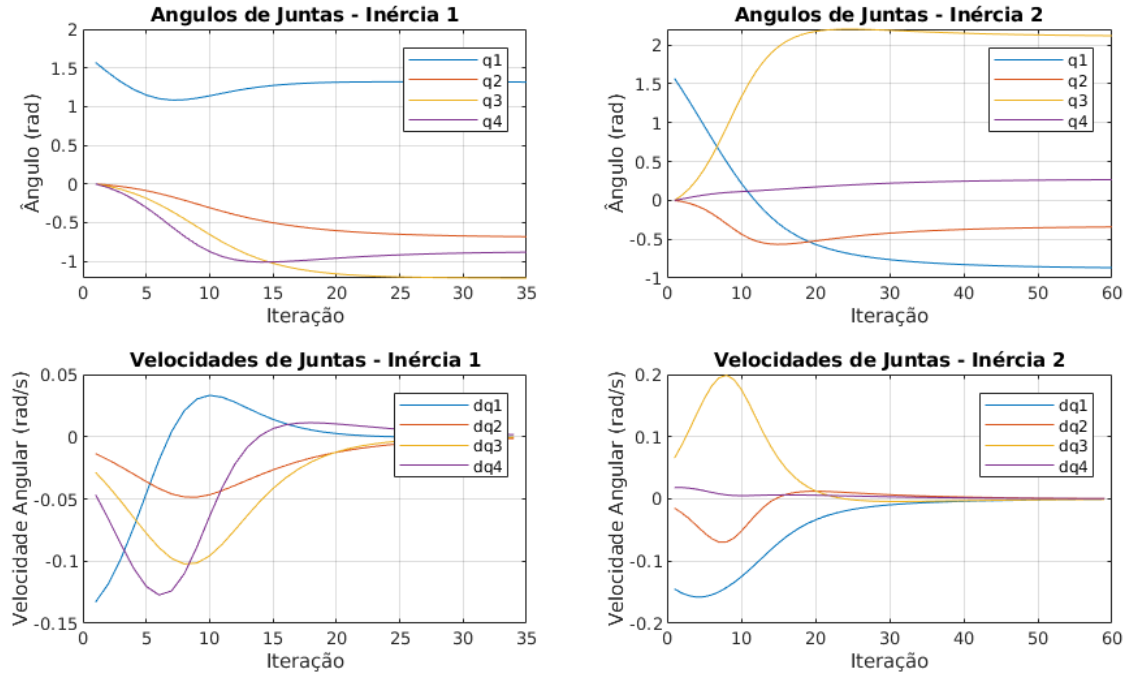


Figura 5.2: Evolução dos ângulos de juntas e das velocidades de juntas ao longo do tempo.

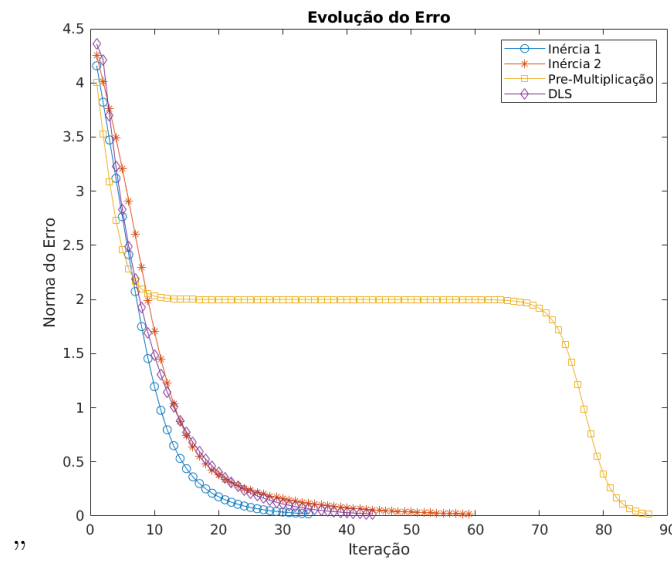


Figura 5.3: Evolução do erro de efetuador ao longo do tempo.

De acordo com o gráfico de erro, o algoritmo que exibiu mais rápida convergência foi o que utilizou a matriz de inércia do primeiro caso (Inércia 1), com 34 iterações, seguido pelo DLS e então pelo segundo caso da matriz de inércia. O comportamento do método da Pré-Multiplicação não foi satisfatório, o que era esperado e foi o que motivou essa escolha de pontos de teste.

5.2 Caso Espacial

O objetivo deste teste é verificar o funcionamento e a competitividade da metodologia desenvolvida neste trabalho em comparação com o método mais comum, o DLS. Para isto, foi utilizado o modelo de um robô similar ao presente no laboratório, cujos parâmetros D-H são mostrados na Tabela 5.2.

Tabela 5.2: Parâmetros D-H para o robô de 7 gDL.

i	1	2	3	4	5	6	7
a_i	0	1	1	1	1	1	1
α_i	$\pi/2$	$-\pi/2$	$\pi/2$	$-\pi/2$	$\pi/2$	$\pi/2$	0
d_i	1	0	0	0	0	0	0
θ_i	q_1	q_2	q_3	q_4	q_5	$\frac{\pi}{2} + q_6$	q_7

Neste teste, foram utilizadas as mesmas constantes k , λ e M do teste planar, e o parâmetro de inércia k_I foi de $1/2$ para todos os elos. A tarefa realizada foi uma simulação de coleta e entrega de objeto, com dois pontos de destino $\mathbf{pDes}_1 = [2.5, 0, 2.5]^T$ e $\mathbf{pDes}_2 = [-2.5, 1, 3]^T$, saindo de uma posição completamente vertical, como ilustrado na Figura 5.4.

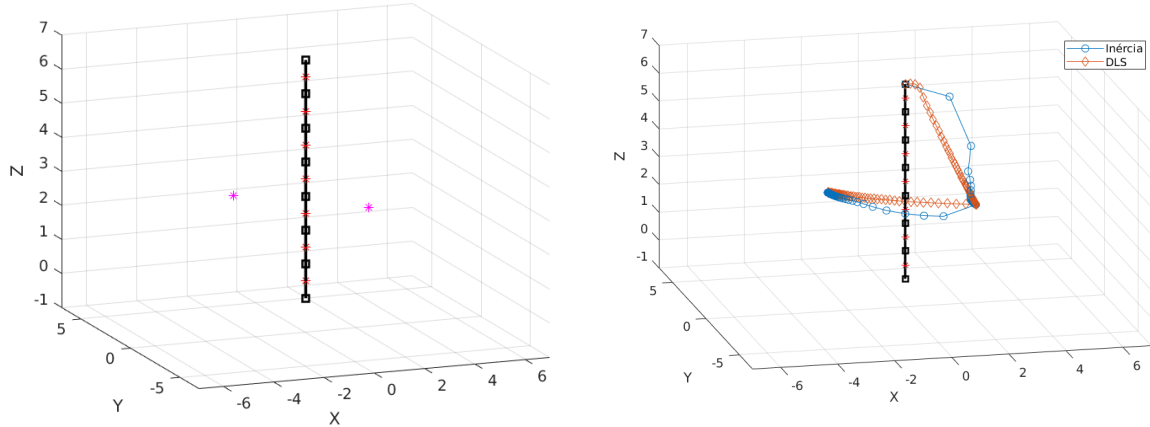


Figura 5.4: Tarefa a ser executada, com pontos de destino em magenta, e trajetória realizada pelos dois algoritmos.

A trajetória realizada pelo efetuador do método desenvolvido se mostrou menos retilínea que o DLS, mas não apresentou as instabilidades características do Jacobiano Transposto, ou seja, a matriz de inércia serviu se mostrou efetiva como substituta da Pré-Multiplicação num robô espacial.

Os dois métodos também podem ser comparados em termos de velocidade de convergência, observando a quantidade de iterações necessárias para alcançar os destinos. Na Figura 5.5, pode-se observar que o método da matriz de inércia convergiu muito mais rapidamente até o destino final, com 71 iterações.

Por ser baseado no Jacobiano Transposto (rápido, porém instável), a metodologia desenvolvida nesse trabalho herdou a característica de velocidade equanto foi capaz de

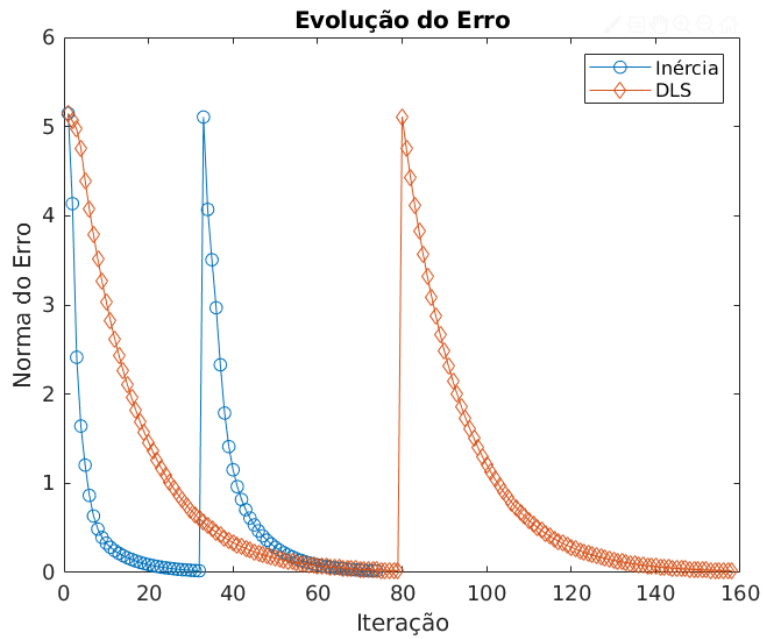


Figura 5.5: Erro de efetuador até os destinos. Os picos são os momentos de alteração do ponto de destino.

controlar a instabilidade que dificulta a utilização do Jacobiano Transposto. Apesar disso, a trajetória mais retilínea foi a do DLS pois é uma melhor aproximação da inversa do Jacobiano, fazendo com que a direção de atuação seja consistente com o vetor de erro.

Capítulo 6

Considerações Finais

Este trabalho foi desenvolvido no intuito de implementar um controlador cinemático considerando aspectos inerciais do robô, representados pela Matriz de Inércia. Ao implementar a metodologia de cálculo dessa matriz, foi possível investigar diferentes situações e o impacto do uso de diferentes valores para os parâmetros do controlador, neste caso massas e inércias virtuais dos elos.

Apesar das relações dinâmicas usadas neste trabalho serem muito investigadas e até expandidas em outros trabalhos, geralmente é feita associação com o Jacobiano Dinamicamente Consistente em sua completude, composto por um par de pré e pós multiplicações do Jacobiano Transposto, que possui problemas de singularidades na pós-multiplicação. Por conta disso, foi desenvolvido um novo controlador que mantém apenas a parte de imune a singularidades do Jacobiano Dinamicamente consistente, que apresentou resultados positivos em termos de estabilidade e velocidade de convergência.

Num trabalho anterior [4], também foi desenvolvido um controlador imune a singularidades baseado numa pré-multiplicação do Jacobiano Transposto por uma matriz específica, resultado de operações sobre uma Rede Neural que representava a cinemática do robô. Neste trabalho foi descoberto que essa matriz é um caso específico da Matriz de Inércia quando a massa dos elos é zero.

Comparando o método desenvolvido com o DLS, este apresenta trajetórias mais retilíneas, mas a ponderação pela Matriz de Inércia atingiu menores tempos de convergência, sem apresentar as instabilidades características do Jacobiano Transposto. A partir dos testes realizados, principalmente no caso de menor dimensão (robô planar), foi possível visualizar a influência que os parâmetros de inércia têm sobre o comportamento do controlador, não afetando a trajetória significativamente, mas alterando a forma como o robô a executava. Intuitivamente, incrementar o valor de inércia de um elo o deixa mais "pesado", e essa noção pode ser usada para alterar o comportamento do manipulador de forma mais intuitiva, o que não é possível em métodos como o DLS.

Em trabalhos futuros, novos aspectos relativos à Dinâmica podem ser considerados, como otimização de acelerações ou torques.

6.1 Atividades Complementares

No período inicial de desenvolvimento deste trabalho, o aluno participou como tutor, árbitro e organizador no V Campeonato de Futebol de Robôs por Simulação de Sergipe, organizado pelo Grupo de Pesquisa em Robótica da UFS (GPR-UFS). Compareceu também ao LARC-CBR 2018 em novembro de 2018, no campeonato de futebol de robôs na categoria *Very Small Size - VSS*, representando o time Lambe Sujo, da Universidade Federal de Sergipe. Durante o evento de futebol de robôs, aconteceu também o LARS 2018, em que foram apresentados os resultados dos trabalhos anteriores [4, 15].

Referências Bibliográficas

- [1] Oussama Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal on Robotics and Automation*, 3(1):43–53, 1987.
- [2] Bruno Siciliano. Kinematic control of redundant robot manipulators: A tutorial. *Journal of intelligent and robotic systems*, 3(3):201–212, 1990.
- [3] Stefano Chiaverini, Bruno Siciliano e Olav Egeland. Review of the damped least-squares inverse kinematics with experiments on an industrial robot manipulator. *IEEE Transactions on control systems technology*, 2(2):123–134, 1994.
- [4] Raphael Cardoso. Movimentação autônoma de robôs manipuladores baseada em técnicas de controle inteligente, (2016).
- [5] Mark W Spong, Seth Hutchinson e Mathukumalli Vidyasagar. *Robot modeling and control*, volume 3. Wiley New York, 2006.
- [6] John J Craig. *Introduction to robotics: mechanics and control*, volume 3. Pearson Prentice Hall Upper Saddle River, 2005.
- [7] Chang Tai Kiang, Andrew Spowage e Chan Kuan Yoong. Review of control and sensor system of flexible manipulator. *Journal of Intelligent & Robotic Systems*, 77(1):187–213, 2015.
- [8] Daniel E Whitney. Resolved motion rate control of manipulators and human prostheses. *IEEE Transactions on man-machine systems*, 10(2):47–53, 1969.
- [9] JOHNM Hollerbach e Ki Suh. Redundancy resolution of manipulators through torque optimization. *IEEE Journal on Robotics and Automation*, 3(4):308–316, 1987.
- [10] Kyong-Sok Chang e Oussama Khatib. Manipulator control at kinematic singularities: A dynamically consistent strategy. In *iros*, page 3084. IEEE, 1995.
- [11] Bojan Nemec e Leon Zlajpah. Null space velocity control with dynamically consistent pseudo-inverse. *Robotica*, 18(5):513–518, 2000.

- [12] Joanna Ratajczak e Krzysztof Tchoń. Dynamically consistent jacobian inverse for mobile manipulators. *International Journal of Control*, 89(6):1159–1168, 2016.
- [13] Luis Sentis e Oussama Khatib. Control of free-floating humanoid robots through task prioritization. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 1718–1723. IEEE, 2005.
- [14] Joanna Ratajczak. Design of inverse kinematics algorithms: extended jacobian approximation of the dynamically consistent jacobian inverse. *Archives of Control Sciences*, 25(1):35–50, 2015.
- [15] Raphael Cardoso. Desenvolvimento de modelos cinemáticos de manipuladores utilizando redes neurais adaptadas, (2017).